



Universidade Federal de Alagoas (UFAL)  
Campus Arapiraca



# Programação Orientada a Objetos (POO)

## 02 - Programação estruturada com Java (Parte 2)

**Alexandre de Andrade Barbosa**

`alexandre.barbosa@arapiraca.ufal.br`

# Objetivos

## Objetivos

- Apresentar as estruturas de controle existentes em Java

# Programação estruturada com Java (Parte 2)

## Estruturas de controle

- As estruturas de controle em Java são as mesmas presentes em C
- A sintaxe das estruturas de controle de Java é igual a sintaxe de C
  - if, if-else (Condicional)
  - switch (Condicional)
  - operador condicional (Condicional)
  - for (Repetição)
  - while (Repetição)
  - do-while (Repetição)

# Programação estruturada com Java (Parte 2)

## Estruturas de condição

### Sintaxe:

```
1  if ( [expressão condicional] ) {  
2      [comandos]  
3  }
```

### Exemplo:

```
1  ...  
2  if (idade < 18) {  
3      System.out.println("Menor de idade!");  
4  }  
5  ...
```

# Programação estruturada com Java (Parte 2)

## Estruturas de condição

### Sintaxe:

```
1 if ( [expressão condicional] ) {  
2     [comandos]  
3 } else {  
4     [comandos]  
5 }
```

### Exemplo:

```
1 ...  
2 if (num % 2 == 0) {  
3     System.out.println(num + ' ' é par!');  
4 } else {  
5     System.out.println(num + ' ' é ímpar!');  
6 }  
7 ...
```

# Programação estruturada com Java (Parte 2)

## Estruturas de condição

### Sintaxe:

```
1  [variável]=[expressão lógica]?[expressão 1]:[expressão  
   2];
```

### Exemplo:

```
1  String n = (num % 2 == 0) ?  
2      num + ' ' é par! '':  
3      num + ' ' é ímpar! '';
```

# Programação estruturada com Java (Parte 2)

## Estruturas de condição

Sintaxe:

```
1 switch ( [expressão] ) {  
2     case [constante 1]:  
3         [comandos]  
4     case [constante 2]:  
5         [comandos]  
6     default :  
7         [comandos]  
8 }
```

# Programação estruturada com Java (Parte 2)

## Estruturas de condição

Exemplo:

```
1  ...  
2  public static void main(String[] args) {  
3      String semestre =  
4          JOptionPane.showInputDialog("Escreva o nome  
5          de um mês:");  
6      semestre = semestre.toLowerCase();  
7      switch (semestre) {  
8          case "janeiro" :  
9          case "fevereiro" :  
10         case "março" :  
11         case "abril" :  
12         case "maio" :  
13         case "junho" :  
14             JOptionPane.showMessageDialog(null ,  
                "Primeiro semestre");  
            break;  
            case "julho" :
```



# Programação estruturada com Java (Parte 2)

## Estrutura de repetição

### Sintaxe:

```
1 for ([inicialização]; [condição de repetição];  
    [incrementos]) {  
2     [sequência de comandos]  
3 }
```

### Exemplo:

```
1 ...  
2 for (int i = 0; i < 30; i++) {  
3     System.out.println(i);  
4 }  
5 ...
```

# Programação estruturada com Java (Parte 2)

## Estrutura de repetição

### Sintaxe:

```
1 while (condição) {  
2     [sequência de comandos];  
3 }
```

### Exemplo:

```
1 ...  
2 int i = 0;  
3 while (i < 30) {  
4     System.out.println(i);  
5     i++;  
6 }  
7 ...
```

# Programação estruturada com Java (Parte 2)

## Estrutura de repetição

### Sintaxe:

```
1  do {  
2      [sequência de comandos];  
3  } while (condição);
```

### Exemplo:

```
1  ...  
2  int i = 0;  
3  do {  
4      System.out.println(i);  
5      i++;  
6  } while (i < 30);  
7  ...
```

# Programação estruturada com Java (Parte 2)

## Funções

### Sintaxe:

```
1 static [tipo de retorno] [nome] ( [argumentos] ) {  
2     [comandos];  
3     return [valor de retorno];  
4 }
```

### Exemplo:

```
1 ...  
2 static boolean paridade (int numero) {  
3     return numero % 2 == 0; // retorna um valor booleano  
4 }  
5 ...
```

Para criar funções sem valor de retorno deve-se utilizar a palavra reservada "void" em

# Programação estruturada com Java (Parte 2)

## Exercícios

### Exercício

- ❶ Crie programas em Java para:
  - ❶ determinar o maior valor entre dois números fornecidos pelo usuário
  - ❷ retornar o valor absoluto de um número fornecido pelo usuário
  - ❸ ordenar três valores fornecidos pelo usuário
  - ❹ ler dois números inteiros  $m$  e  $n$ , e imprimir os números pares existentes no intervalo  $[m, n]$
  - ❺ ler uma sequência de números, até que o usuário digite 0 (zero), e apresentar o maior valor fornecido

# Programação estruturada com Java (Parte 2)

## Exercícios

### Exercício

- 1 Exercícios das páginas 72 e 73 do livro “Java 6 - Ensino Didático” (Estruturas de controle)
- 2 Exercícios das páginas 113 e 114 do livro “Java 6 - Ensino Didático” (Métodos)

# Programação estruturada com Java (Parte 2)

## Resumo

### Resumo

- Estruturas de controle estão presentes em todas as linguagens imperativas (ex. estruturas e orientadas a objetos)
- A sintaxe pode variar de uma linguagem para outra, a semântica é "sempre" a mesma
- Java possui as seguintes estruturas de controle
  - if, if-else (Condicional)
  - switch (Condicional)
  - operador condicional (Condicional)
  - for (Repetição)
  - while (Repetição)
  - do-while (Repetição)

# Programação estruturada com Java (Parte 2)

## Resumo

```
1 if ( [condição] ) {  
2     [comandos];  
3 } else {  
4     [comandos];  
5 }
```

```
1 [variável] = [condição] ?  
2     [expressão 1] :  
3     [expressão 2];
```

```
1 switch ( [expressão] ) {  
2     case [constante 1]:
```

```
1 while ( [condição] ) {  
2     [comandos];  
3 }
```

```
1 do {  
2     [comandos];  
3 } while ( [condição] );
```

```
1 for ( [inicialização];  
2     [condição de  
3     repetição];  
4     [incrementos] ) {
```



# Programação estruturada com Java (Parte 2)

## Exercícios

### Exercício

- 1 Observe os exercícios no sistema

# Perguntas?

**Alexandre de Andrade Barbosa**  
*[alexandre.barbosa@arapiraca.ufal.br](mailto:alexandre.barbosa@arapiraca.ufal.br)*